



**PAK-AUSTRIA FACHHOCHSCHULE:**  
**INSTITUTE OF APPLIED SCIENCES AND TECHNOLOGY**

### **Data Structures & Algorithms Project Proposal**

#### **Group Members**

No.	Name	Registration No.
1.	<b>Ahmad Ghani</b>	B24F0356AI079
2.	<b>Hidayat Ullah</b>	B24F0050AI099
3.	<b>Umer Farooq</b>	B24F0418AI076
4.	<b>Ibrahim Khan</b>	B24F2174AI075
5.	<b>Aswerah Rehman</b>	B24F1020AI098

**Pak-Austria Fachhochschule: Institute of Applied Sciences and Technology**

**(PAF-IAST)**



# **University Management System (CLI-Based)**

## **Project Overview**

The University Management System (UMS) is a Command Line Interface (CLI)-based application designed to manage and organize important university records efficiently. The system focuses on three major modules: Student Record Management, Exam Record Management, and Transport Record Management. The goal of this project is to demonstrate how Data Structures and Algorithms (DSA) can be used to efficiently store, retrieve, update, and manage data in different modules of a university system.

## **Project Objectives**

- To design a simple and efficient system for managing university data.
- To apply data structures (arrays, linked lists, queues, stacks, trees, and files) in a practical project.
- To improve searching, sorting, and record-handling operations.
- To enhance understanding of modular programming and data handling using C++.

## **System Modules and Features**

### **1. Student Record Management**

Purpose: Maintain all information related to students.

Features:

- Add new student record
- Display all student records
- Search student by ID or name
- Update student details (e.g., department, phone number)
- Delete student record
- Save and load data from a file

Data Structure Used: Linked List or Array, File Handling, Sorting & Searching algorithms.

### **2. Exam Record Management**

Purpose: Manage student exam details and grades.

Features:

- Add exam results (student ID, course code, marks, grade)
- View all exam records
- Search results by student ID or course
- Calculate GPA or average marks



- Sort records by marks or student name
- Update or delete exam entries

Data Structure Used: Array of Structures / Linked List, Sorting and Searching Algorithms.

### **3. Transport Record Management**

Purpose: Maintain transport details for students using university transport.

Features:

- Register student for transport
- Assign bus route and driver
- Display all transport records
- Search by route number or student ID
- Update transport record
- Remove a student from transport list

Data Structure Used: Queue, Linked List / Array, File Handling.

### **Core DSA Concepts Used**

- Array: Storing fixed-size data like exam marks
- Linked List: Managing dynamic student records
- Queue: Managing transport requests
- Stack: Undo/redo operations (optional)
- Sorting Algorithms: Organizing records by name, marks, or ID
- Searching Algorithms: Finding student, transport, or exam record
- File Handling: Saving and loading data permanently

### **Expected Output / Functionality Flow**

1. Main Menu:

===== UNIVERSITY MANAGEMENT SYSTEM =====

1. Student Record Management
2. Exam Record Management
3. Transport Record Management
4. Exit

Enter your choice:

2. Each module opens a submenu (e.g., Add, Display, Search, Delete).
3. Data is processed using chosen data structures and saved to files for persistence.
4. Output is displayed in a structured tabular format in the terminal.



## **Tools & Technologies**

- Programming Language: C++
- Compiler: g++ / Code::Blocks / Visual Studio Code
- Platform: Windows or Linux (CLI-based)

## **Future Enhancements**

- Convert the CLI system into a GUI or web-based version.
- Add authentication (admin login).
- Integrate database (MySQL / SQLite).
- Include attendance and fee management.

## **Expected Learning Outcomes**

- Improved understanding of data structures and their real-world applications.
- Experience with file handling, modular coding, and algorithm efficiency.
- Hands-on practice in designing CLI-based management systems.

